Example evaluation project in C#

XML serializer

This project specification is intended only for the use by the designated candidate. It is confidential and privileged. If you are not the named recipient for a developer position, you are hereby notified that any unauthorized review, disclosure, copying, or use of the content of this transmission is prohibited. Please do not share this specification around.

The purpose of this project is to demonstrate one's abilities to implement a small solution in C#.

Specification

Please build two projects together with automated database migrations.

The first project would be an API with Swagger and two endpoints:

- **POST** /api/data - this endpoint receives a collection of serialized JSON models and stores them in a database created using automated database migrations.

The migrations project needs to create only one table in a local database to store the models with the columns matching the properties.

Table name:

```
dbo.Requests
```

The request objects in JSON should have the following structure:

```
{
  "ix": INT,
  "name": STRING,
  "visits": INT?,
  "date": DATETIME
}
```

The equivalent of the following in C#

```
public class RequestModel
{
    public int Index {get;set;}
    public string Name {get;set;}
    public int? Visits {get;set;}
    public DateTime Date {get;set;}
}
```

- **GET** /api/jobs/saveFiles - this endpoint invokes an internal job of selecting the database records and then serializes each table record as an XML file. The files are saved in App_Data\xml\yyyy-MM-dd based on the date value in the models in the following format:

```
<?xml version="1.0" encoding="UTF-8"?>
<request>
  <ix>ID</ix>
  <content>
    <name>NAMEVALUE</name>
    <visits>INT</visits>
    <dateRequested>yyyy-MM-dd</dateRequested>
  </content>
</request>
```

For Visits being null, please serialize to this format

```
<?xml version="1.0" encoding="UTF-8"?>
<request>
  <ix>ID</ix>
  <content>
    <name>NAMEVALUE</name>
    <dateRequested>yyyy-MM-dd</dateRequested>
  </content>
</request>
```

2. The second project would be a console application which sends a collection of serialized Request models via a HTTP POST request to the API. The address of the

API should be configured in the App.config file. The console application should receive one input parameter from the command line which would be an INT which specifies the number of models from point number 1. in the array to send. Please use https://github.com/bchavez/Bogus to generate random values for these models as the data to send to the API.

Technical requirements

- C#
- WebAPI2
- Swagger
- Autofac
- xUnit
- NSubstitute
- dbUp, Simple.Migrations etc
- Unit tests
- Integration tests
- Database tests using migrations
- High test coverage